

` 1

Getting Started with RabbitMQ on Windows

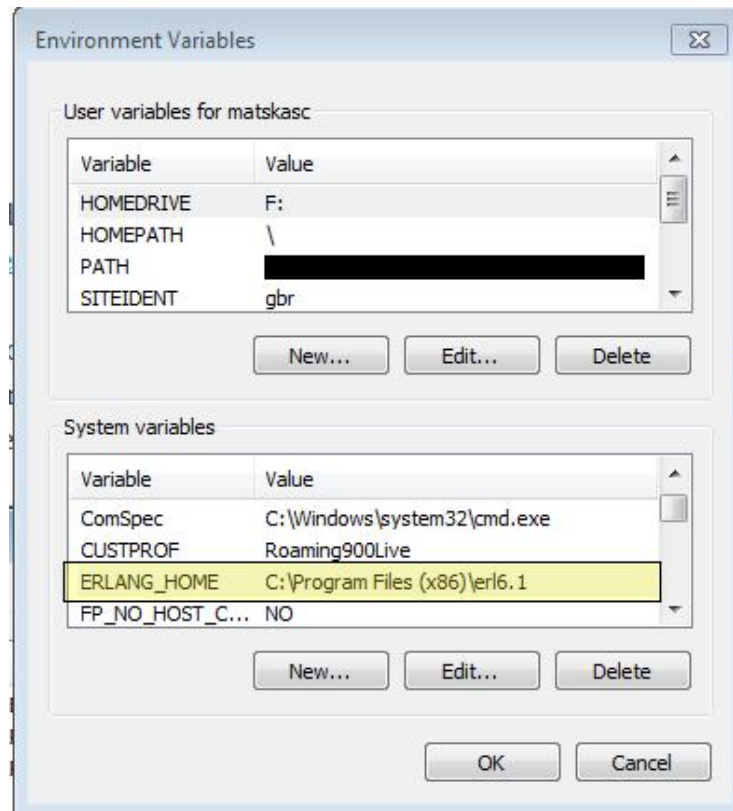
28 July 2014 2 Comments ([/getting-started-with-rabbitmq-on-windows/#disqus_thread](#)) Posted in [.NET \(/tag/net/\)](#), [Windows Service \(/tag/windows-service/\)](#), [Queue \(/tag/queue/\)](#)

RabbitMQ is a great queuing tool that can run on a variety of platforms. If you are planning on doing any distributed, asynchronous processing and you need a system to queue and process items in the background, then you should seriously consider RabbitMQ. The online documentation is great and there are plenty of tutorials that can take you through the basics of creating and consuming queues in your application. However, before you start typing any code, you first you need to get RabbitMQ up and running somewhere (local system, server etc). In this post, we will go through the installation steps and see what's needed to get us to that point.

1. ERLANG INSTALLATION

RabbitMQ runs on the Erlang virtual runtime. Don't ask me why, but it does, so we need to install Erlang first as without it we can't get RabbitMQ to work. You can get the latest Windows installer here (<http://www.erlang.org/download.html>). Make sure you choose the 32 or 64bit version depending on the system you plan to run it. Download and install it. There is no restart required, so this first step is easy. Before we continue, ensure that the appropriate environment variable (`ERLANG_HOME`) has been

created during the installation. If, for any reason, the environment variable is missing, you will need to create it manually as per the image below:



2. RABBITMQ SERVICE INSTALLATION

Grab the latest installer for Windows from the RabbitMQ website here (<http://www.rabbitmq.com/install-windows.html>) Download and install. Again, nothing fancy here.

3. RABBITMQ WEB MANAGEMENT PLUGIN INSTALLATION

RabbitMQ runs, by default, as a Windows Service and technically you shouldn't need to do anything else to start interacting with it. However, any interaction with the service needs to happen over the command-line. If you are not a die-hard command-line, then

there is an solution for you. RabbitMQ comes with a handy and, nice if I may say, web-based management plugin which is part of the installation files. To install the web plugin you just need follow the steps below:

- Open an **elevated** command line (Run as Administrator)
- Navigate to the **sbin** directory of the RabbitMQ Server installation directory. In my case the path is `C:\Program Files (x86)\RabbitMQ Server\rabbitmq_server-3.3.4\sbin`
- Run the following command to enable the plugin `rabbitmq-plugins.bat enable rabbitmq_management`
- Then, re-install the RabbitMQ service using the commands below:

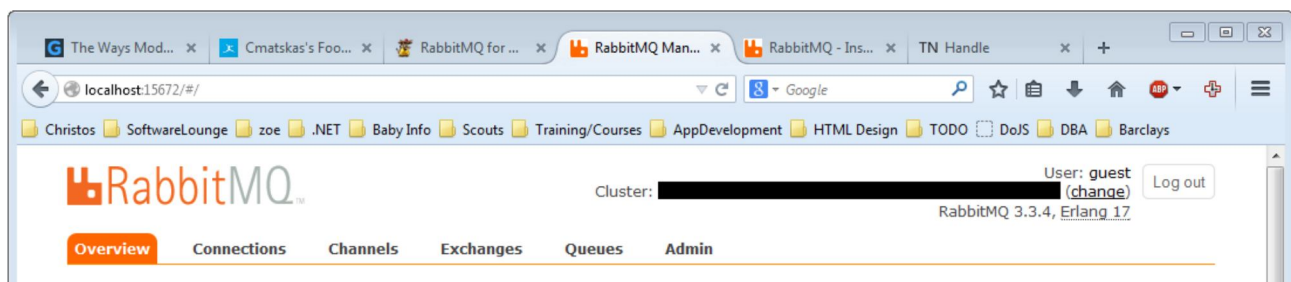
```
rabbitmq-service.bat stop
rabbitmq-service.bat install
rabbitmq-service.bat start
```

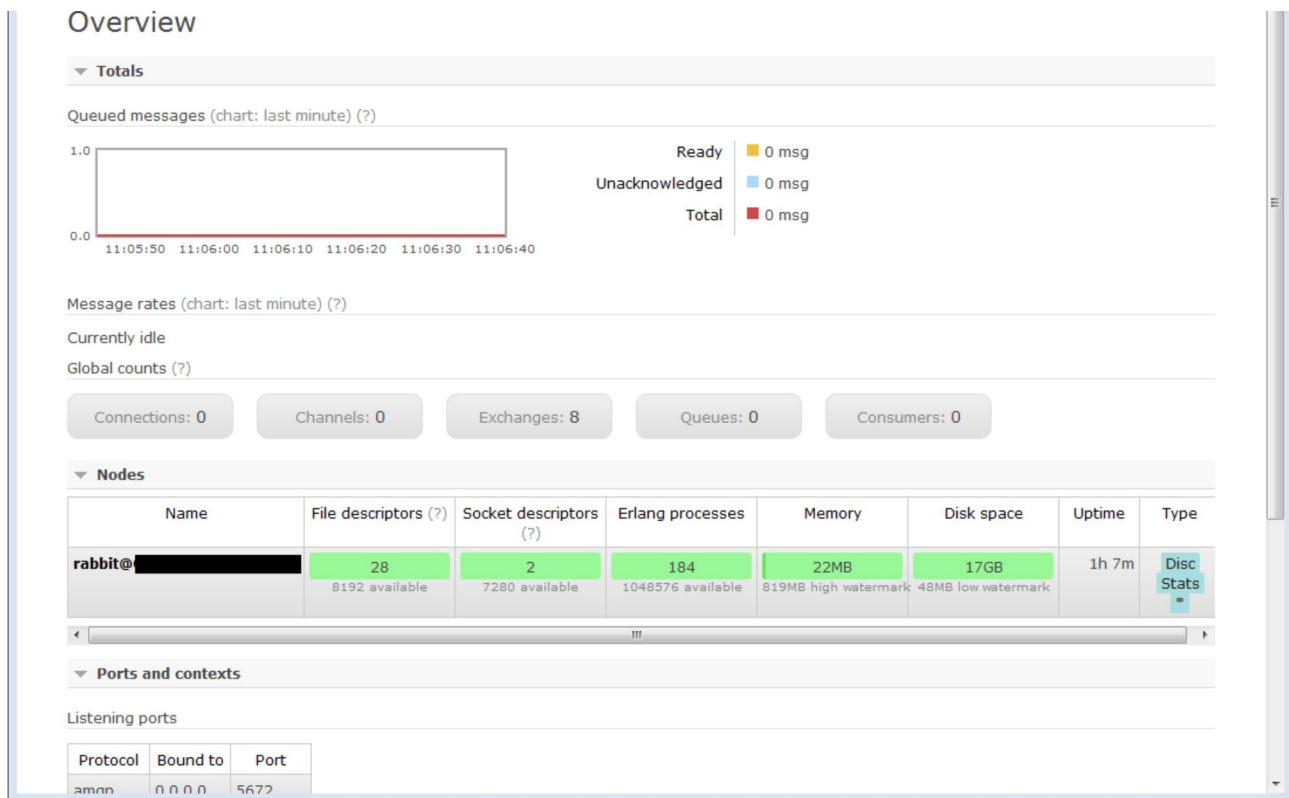
To check if everything worked as expected, navigate to `http://localhost:15672/mgmt` (`http://localhost:15672/mgmt`). You will be prompted for username and password. The default credentials are:

Username: **guest**

Password: **guest**

You should now be presented with the following page:





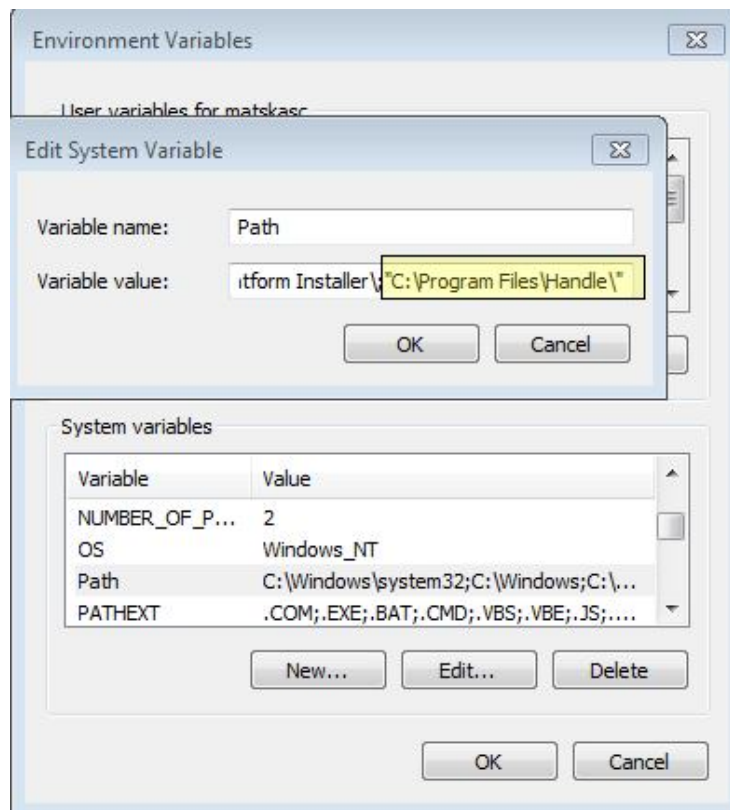
4. INSTALL HANDLE.EXE FOR RABBITMQ'S FILE DESCRIPTORS

When you first log in to the RabbitMQ web dashboard, under the "Overview" tab you will notice that in the "File descriptors" field there is a question mark about the missing **Handle.exe**. **Handle.exe** enables RabbitMQ to monitor the local file system. It will still work without it but I think that it's better to make it available, so let's go and get it. The dashboard provides a handy link to the Handle download page, but you can also use this link (<http://technet.microsoft.com/en-us/sysinternals/bb896655>) to go to that same page directly. As a side-note, this tool is written by no one else but Mark Russinovich himself. This guy is a genius!

Download the zip file and unzip it anywhere you want. I decided to extract mine in "C:\Program Files\Handle". It is important that you

keep a note of the extracted path as we will need to add it to **PATH** environment variable so it is available to RabbitMQ.

- Open the Environment Variables Windows,
- Scroll down to the System Variables -> Path variable and click on Edit.
- At the end of the Variable Value, add the path to the Handle.exe:



- Click OK twice to close all windows down

Finally, we need to restart the RabbitMQ service so that our changes can be picked up. Use the steps below to restart the service:

- Open up an **elevated** command line
- Type the following commands

```
rabbitmq-service.bat stop  
rabbitmq-service.bat start
```

Go back to the web dashboard and check the Overview page. The warning about the missing Handle.exe should now be gone.

Next time we will look into how we can use the queuing platform and integrate it with our .NET code.


Happy coding...


P.S. Make sure you follow me on Twitter @christosmatskas (<https://twitter.com/christosmatskas>) for more up-to-date news, articles and tips.

Share this post on

 (<https://twitter.com/christosmatskas>)

[/share?text=Getting%20Started%20with%20RabbitMQ%20on%20Windows
url=https://cmatskas.com/getting-started-with-rabbitmq-on-windows](https://twitter.com/christosmatskas)

 (<https://www.facebook.com/sharer/sharer.php?u=https://cmatskas.com/getting-started-with-rabbitmq-on-windows/>)

 (<http://www.linkedin.com/shareArticle?mini=true&url=https://cmatskas.com/getting-started-with-rabbitmq-on-windows/&title=Getting%20Started%20with%20RabbitMQ%20on%20Windows&source=cmatskas.com>)

 (<https://plus.google.com/share?url=https://cmatskas.com/getting-started-with-rabbitmq-on-windows/>)


 An introduction to the Common Logging API for .NET ([an-introduction-to-common-logging-api-2/](/an-introduction-to-common-logging-api-2/))

[Blog Home \(https://cmatskas.com\)](https://cmatskas.com)

An introduction to JavaScript Unit Testing with qUnit and Sinon.js ➔ (/an-introduction-to-javascript-unit-testing-with-qunit-and-sinon-js/)

4 Comments

cmatskas

 Login ▾ Recommend 2 Share

Sort by Best ▾



Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS **Sandeep Chougule** • a month ago

Thank you so much it helped.

  • Reply • Share ›**Some Guy Somewhere** • 2 years ago

How can one adjust the logging level at runtime? Say, in production, we want to run for an hour at trace level, without changing the config file and bringing the site offline?

  • Reply • Share ›**Dan Beaulieu** • 2 years ago

Visual Studio says GetCurrentClassLogger is deprecated, not sure how to proceed. Any thoughts?

  • Reply • Share ›**Christos Matskas** **Mod** → Dan Beaulieu • 2 years ago

Hi @DevelopmentCenter thanks for reading and for your comment. I've updated the code now to use the later API (3.3.0) so I would suggest you follow my code example when instantiating a local Logger. I hope this helps.

1   • Reply • Share ›

ALSO ON CMATSKAS

Install and run the Azure CLI on the Windows Subsystem for Linux (WSL)

2 comments • 10 months ago

**Steve Crabb** — It's worth noting that you need to install Node.js/npm on WSL as it's not installed by default:sudo -i apt-get install**Working for Microsoft as a Premier Field Engineer**

3 comments • 8 months ago

**Shane** — So reading between the lines its me you enjoy working with the most :) Very inspirational post !**Configure a RunOnce task on Windows**

6 comments • a year ago

**Darwin IS** — Would be nice to correct your final**Deploying Functions with ARM templates and inline code**

2 comments • 10 months ago

